TEXTURE MAPPING
Texture mapping

At what point do things start looking real?
Texture mapping

- Surfaces have details containing high frequencies
  - Orange is rough
  - Wooden table carries the wood texture
  - Mirrors reflect the environment
- Idea: Modify the shading equation at several places
  - The Color Hue
  - The Color Intensity
  - The Surface Normal
Linear Texture Patterns

- Look up table (LUT) with index $s \in [0, 1]$
- Mapping procedure:
  - Define the color on two spatial positions
  - Linearly modify the color between the two points
  - Interpolate the color value if needed
    - Nearest neighbor, Linear ...

- What do we get?
- What do we get for $s \in [-0.1, 1.1]$?
Surface Texture Patterns

- How can we map the image to the object?
Surface Texture Patterns

Texture Space: (s,t) Texel Coordinates

Object Space: (u,v) Surface Parameters

Image Space: (x,y) Pixel Coordinates

Pattern Array

Surface

Pixel Area
Illumination equation becomes:

\[ I = \sum_i [I_{a_i} K_a + I_{p_i} C(s, t)(K_d \cos(\theta) + K_s \cos^n(\alpha))] \]

- \( C(s, t) \) is the 2D texture array
Parametric Texture 1

- Planar mapping
Parametric Texture 2

- Cylindrical mapping
Parametric Texture 3

- Spherical mapping
Parametric Texture 4

- Box mapping
Parametric Texture 5

- Box mapping for environment mapping
Option: Unfold the Surface

[Piponi and Borshukov 2000]
Option: Atlas

[Sander et al. 2001]
Exercise
Surface Texture Patterns

- **Difficulties:**
  - Aliasing
  - Boundaries between surfaces
  - Isometric mapping
Volume Texture Patterns

- 2D vs. 3D
Volume Texture Patterns

- Map a solid texture $\rho(s, t, r)$ to the scene
- Similar to carving object out of a solid
- Used for natural material or irregular textures (stone, wood, etc...)
- Allows internal view – slices of the solid texture
Volume Texture Patterns

- **Strengths:**
  - Shared boundaries are seamlessly textured
  - Isometric mapping

- **Difficulties:**
  - Not trivial to define
  - Transformations
  - Large storage
Procedural Texturing

- Calculate variations (noise) for the properties of an object
  - Defined for every point on the object space
  - No need in transformations
  - No need in large storage

- Solid Noise
- Turbulence
Solid Noise

\[ n(x, y, z) = \sum_{i=\lfloor x \rfloor}^{\lfloor x \rfloor + 1} \sum_{j=\lfloor y \rfloor}^{\lfloor y \rfloor + 1} \sum_{k=\lfloor z \rfloor}^{\lfloor z \rfloor + 1} \Omega_{ijk}(x - i, y - j, z - k) \]

\[ \Omega_{ijk}(u, v, w) = \omega(u)\omega(v)\omega(w) \langle R_{ijk} \cdot (u, v, u) \rangle \]

\[ \omega(t) = 2|t|^3 - 3|t|^2 + 1, \quad |t| < 1 \]

- \( R_{ijk} \) is a random vector on unit square
Turbulence

function turbulence(p)
    t = 0;
    scale = 1;
    while (scale > pixelsize) {
        t += abs(Noise(p/scale)*scale);
        scale /= 2;
    }

    return t;
Turbulence

\[ g_1(x) := \sin(x) \]

\[ g_2(x) := \sin(x) + \frac{1}{2} \sin(2x) + \frac{1}{4} \sin(4x) + \frac{1}{8} \sin(8x) + \frac{1}{16} \sin(16x) + \frac{1}{32} \sin(32x) \]
Turbulence
Turbulence
Bump Mapping

- Change object normals before rendering
- Both magnitude and direction are subject to change
- The geometry does not change

No bump mapping

With bump mapping
Bump Mapping

- If transition is from white to black, rotate the point’s surface normal by $\theta$
- If transition is from black to white, rotate the point’s surface normal by $-\theta$
Displacement Mapping

- Alters object geometry
- Displaces points along normal

Bump Map  Displacement Map  Bump Map  Displacement Map